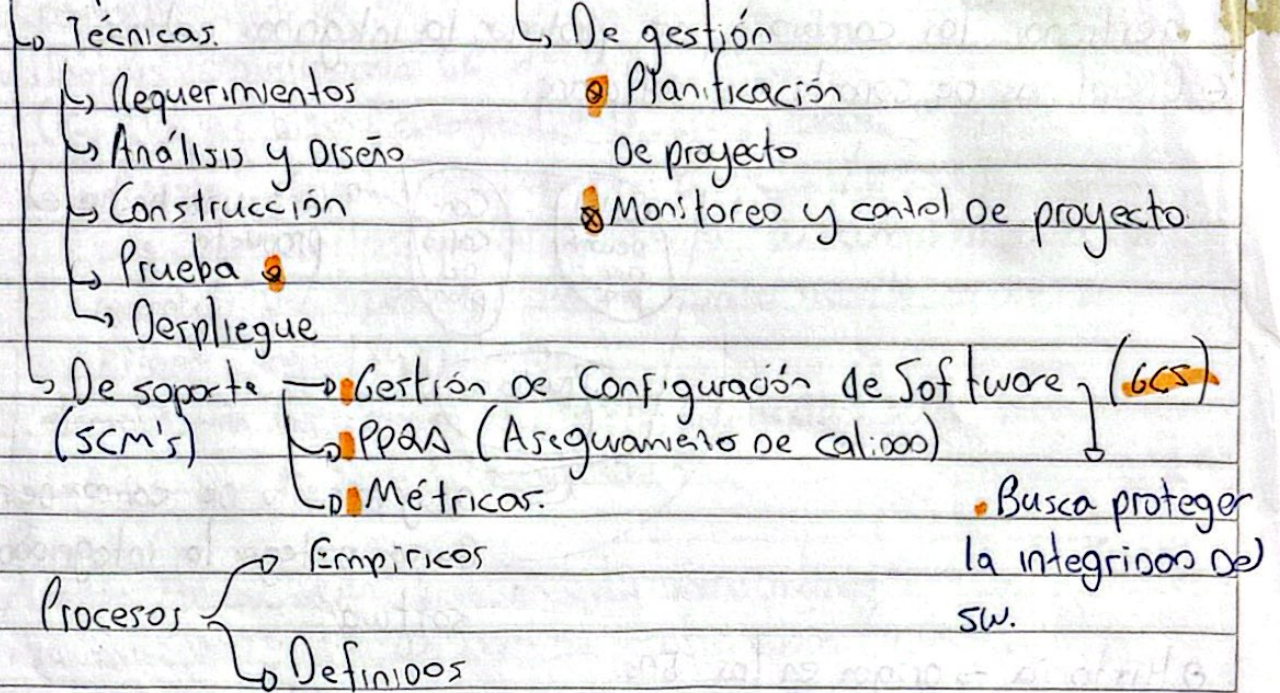
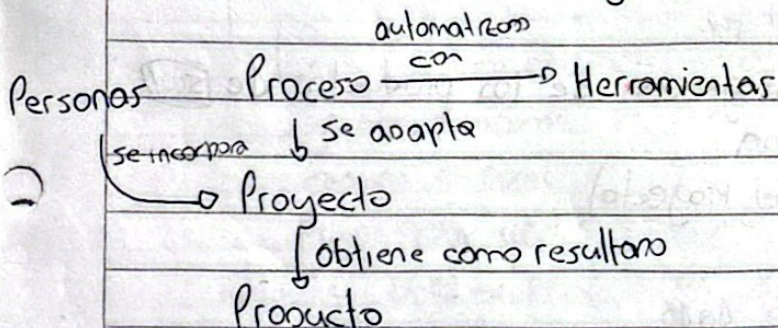


Disciplinas del desarrollo de sw.



SCM (Software Configuration Management)



El software. → Definición

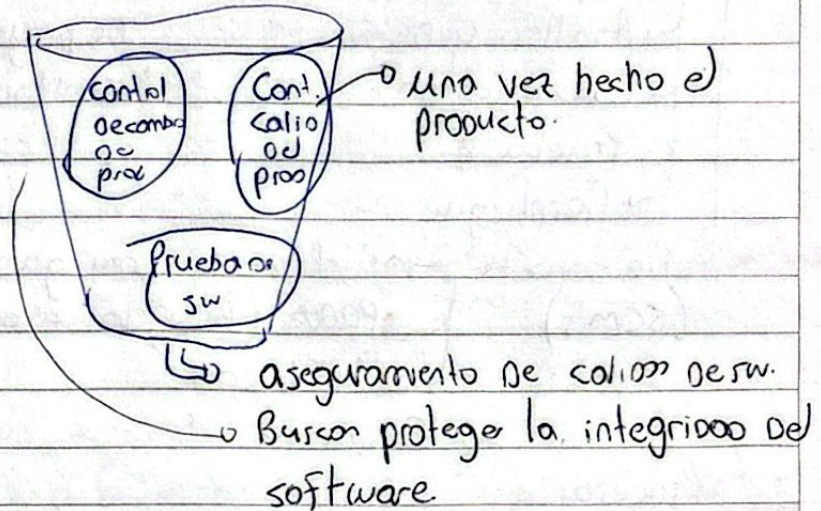
Cambios en el software

- origenes
 - Cambios en el negocio y nuevos req's.
 - Soporte de cambios de productos.
 - Reorganización de prioridades de la empresa por crecimiento.

⊖ SCS como disciplina de soporte.

↳ Es una actividad "paraguas", transversal a todo el proyecto, relevante para el producto a lo largo de su ciclo de vida. La idea es gestionar los cambios para proteger la integridad del sw.

⊖ Disciplinas de soporte del software.



⊖ Historia → origen en los 50s.

⊖ Definición de la IREB de la ISO 9005

⊖ Propósito de gestionar la configuración de sw.

↳ establecer y mantener la integridad de los productos de sw a lo largo de su ciclo de vida
(Producto, más duradero que el Proyecto)

Involucra la configuración

- ↳ Identificarla en un momento dado
- ↳ Controlar sistemáticamente sus cambios.
- ↳ Mantener su integridad y origen

⊖ Integridad de un producto.

↳ Satisfaga las necesidades del usuario
↳ un prod. es íntegro cuando \rightarrow si el proceso durante su ciclo de vida cumple con exp. de calidad

⊖ Problemas de manejo de componentes

② Conceptos clave de GCS.

↳ Ítem de configuración de Software (SCI)

(IC) Doc. de diseño, código fuente, código ejecutable, etc.

↳ Artefactos que forma parte del proyecto / producto que pueden sufrir cambios o necesitan ser comparados entre los miembros del equipo y sobre los cuales necesito conocer su estado; control y evolución.

Ejemplos → Req's; manual de usuario; Arquitectura del sw; código fuente; estándares de configuración.

↳ Versión: forma de un IC en un determinado momento.

↳ Variante: versión de un ítem de configuración (o de la config) que evoluciona por separado.

③ ¿Qué es una config. de sw? Foto de un conjunto de ítems a una versión dada.

↳ Línea base → es una configuración que los define como íntegro, ya que es como se vuelve cuando se rompe algo y es un punto seguro en donde todo funciona y es una base segura.

Puede tener uno o ④ ítems, es una etiqueta de varios ítems.

miércoles 27/08 → práctico de GCS

Rama ≠ Variante

↳ no necesariamente se junta
↳ se junta y no "son" línea base.

¿Qué es un repositorio es un contenedor de ICs, que mantiene la historia de cada IC con (ext.)

④ Funcionamiento del repo: Tengo 2 acciones principales:
↳ 2 tipos → Centralizados (2 acciones)
↳ descentralizados (c/ cliente tiene una copia exacta del repo completo)

↳ f. fallido igual del repo completo

① Identificación de líneas base.

- ↳ Se utilizan etiquetas para "marcar" la línea base.
- ↳ a versión del producto

② Líneas Base:

③ Representación de líneas base

- ↳ Las utilizo como Planificación, como termino los req. funcionales

④ Creación de ramas

- ↳ existe una rama principal (trunk, main)
- Sirven pl bifurcar el desarrollo.
- Todas las ramas deberán integrarse a la principal o ser descartadas
- Pueden surgir conflictos (se resuelve con diff)

⑤ Definición de GCS.

Plan de gestión

↳ Activos fundamentales de la GCS

↳ Identificación de ítems de configuración.

↳ Ident. unívoca

↳ Convenciones y reglas de nombres

↳ Def. estruc. del Repositorio

↳ Ubicación dentro de la estruc. del repo

⑥ Proceso de

① Control de cambios

② El comité del control de cambios. (para cambios que afecten al producto, que tengan un impacto grande en lo monetario)

③ Auditoría de configuración de SW

↳ Auditoría funcional de config.

↳ Auditoría física

↳ Las realiza un agente externo al proyecto (o un auditor externo)

Auditorías de gestión de config. de sw.: Buscan validar (¿se obtiene el producto) y verificar

↳ Salida (Informe de auditoría con hallazgos, mejoras y oportunidades de mejora)

④ Informes de estado → mantiene los registros de evolución del sistema.

⑤ Registro de informe de estado

⑥ Preguntas con reg. de estado

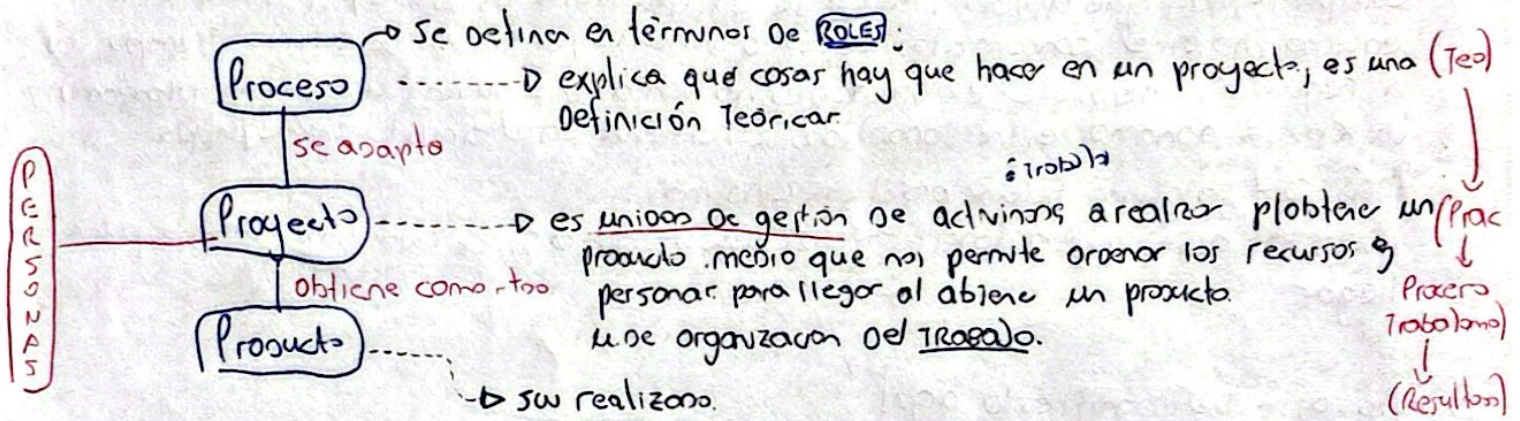
⊖ 6CS para/en ambientes ágiles (lee monitriesto ágil ~ No silver bullet) ⊕

⊖ Scm en Agile

⊖ scm en ágil, algunos tips

⊖ " " " ; Para Debatir { Pld miércoles que viene responder.

Clase 15W 22/08/25. → Filosofía del monitriesto ágil.



⊖ Características de un proyecto

- Resultado único (no hay líneas de producción masiva, ej. vacaciones)
- Esfuerzo temporal definido (Debe haber una fecha inicio a una fecha fin)
- Elaboración gradual. (Descomposición de objetivos a través de tareas interrelacionadas más chicas)

⊖ Proceso

- Definido: Busca la previsibilidad y la visibilidad; hay alguna que define los procesos de la organización (Tiene ⊕ información)
- Empírico: La experiencia no es extrapolable;

gran dif → el definido

3 pilares

① Transparencia

② Inspección

③ Adaptación

①

ciclo de vida ≠ Proceso

→ el producto no es el proyecto

Agile / Lean filosofías
↓
conjunto de conceptos

Referentes
Agile → scrum
Lean → kanban

ciclo de vida o modo de proceso

Tipos: Secuencial (casero) / Iterativo (PaaS - incremental)

Luego de elegir el proceso, elijo de ciclo de vida de vino a usar para estructurar el proyecto, el ciclo de vida me da la idea de cómo en el que hago las act.

2 tipos de procesos.

Definido: Busca un objetivo, la viabilidad y predictibilidad, si definimos detalladamente lo que debemos hacer y se hace lo que se debe hacer, estaremos en condiciones de decir cuánto demora en hacer el sw. Hay un proceso que se debe respetar, es más formal. Al ser un proceso organizacional, hay que justificar el cambio en algún proceso.

Empírico: Dice que la **experiencia no es extrapolable**, la experiencia no se comparte, hay que aprender, la **xp** se complementa en ese equipo, solo en ese equipo (no es el conocimiento técnico o vinculos, es la xp de cuánto tiempo voy a requerir, dinero, etc; ex. vacaciones, no voy a ser = nunca cuando las planeamos) la xp la ganamos entre todos. Todo se resuelve en el ámbito del equipo.

más **soft**: algunas buenas prácticas y no más.

Necesita gente que autogerencie, se debe tener una base de xp. para llegar

Filosofía del manifiesto ágil

Procesos empíricos: Proceso que se basa en lo exp. directa, la obs. y la experimentación. En vez de seguir un plan rígido definido, el conocimiento se deriva de la **xp** y toma en cuenta lo observado para tomar decisiones.

Valores ágiles —o innovarse e interactuar vs Procesos en herramienta
o sw funcionando vs Ocurrencia excesiva ①
o Colaborar con el cliente vs negociaciones contractuales
o responder al cambio vs Seguir un plan.

Los 12 principios del manifiesto ágil:

- ① Nuestra mayor prioridad es satisfacer a los clientes
- ② Aceptar que los req. **cambian**
- ③ Entregar sw funcionando **frecuentemente**
- ④ Los responsables de negocios, diseñadores, desarrolladores, deben **trabajar juntos** día a día durante el proyecto. (Product owner)

- ⑤ Desarrollamos proyectos a torno a **individuos** motivados
- ⑥ el método más eficiente de comunicar entre info es **conversaciones cara a cara**
- ⑦ el sw funcionando es la **principal medida de éxito**
- ⑧ Los procesos ágiles promueven el **desarrollo sostenible**
- ⑨ La atención continua a la **excelencia técnica** y el **buen diseño** mejora la **agilidad**
- ⑩ La **simplicidad** es esencial
- ⑪ Las mejores **arquitecturas, diseños** emerge de **equipos**

⑫ **Atendemos** regulares al **equipo** reflexando sobre **cómo ser más efectivos** no **cómo ser más rápidos**